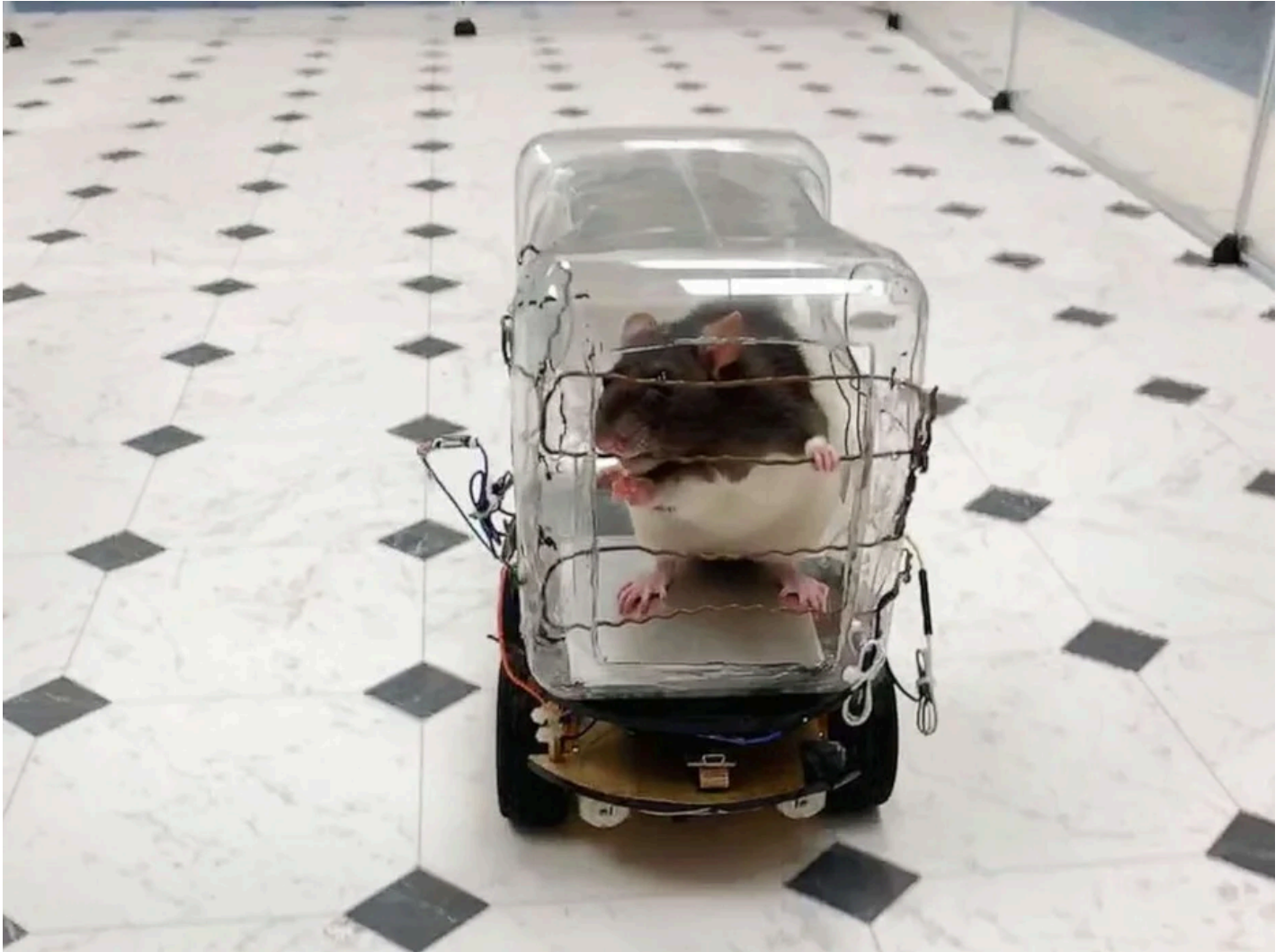# Rat-Operated Vehicle

By lcrawfor in CircuitsArduino

Introduction: Rat-Operated Vehicle

Authors**:** L. E. Crawford & Thad Martin

Introduction

This Instructable teaches you how to build an electric vehicle that can be driven by a rat or other small animal. This car was used for a multidisciplinary research project in which we taught rats how to drive. News coverage of the research and videos are here: https://www.washingtonpost.com/science/2019/10/24...). Published paper is here: https://doi.org/10.1016/j.bbr.2019.112309

*Rat-centered design.*The rat-machine interface is designed to capitalize on a behavior that rats naturally do -- pawing. The car has bars across the front and side windows, and the rat can control the car by grasping a bar with its paw. Touching the bars in front of the car makes it drive forwards, touching the bars on the right side makes it turn right, and touching the bars on the left makes it go left. Touching a front bar and side bar at the same time makes it curve in that direction. Releasing contact with the bars makes the car stop.

How does it work? There is low voltage in the plate that the rat stands on. When the rat touches the copper bars that cross the cars window, the body of the rat acts as an electrical resistor. This raises the voltage of the bars, which is detected by the Arduino. When the Arduino senses an increase in voltage, it controls the motors.

*Is it safe?*The electrical current going from the Arduino and through the rat is about .005 milliamperes. This is orders of magnitude smaller than the minimum current that a human can detect (which is 1 milliampere, according to the National Institute for Occupational Safety and Health). This poses no risk to the rat.

However, there are other risks that you can mitigate. Do not leave any exposed sharp edges or pointy wires on the car because you or the rat could get cut or poked. Also, for good hygiene, you must be able to disassemble and clean any part of the car that the rat touches. Most importantly, NEVER leave a rat unsupervised with the car. It could get hurt by chewing on the wires or doing something else we haven't thought of. Driving is inherently risky and it's on you to keep your rat safe.

Overview of the ROV. We built this car on the platform of an Elegoo Smart Car (V3) kit. Originally intended as a prototype, it's admittedly shabby, but it worked and the rats didn't care about the aesthetics, so we stuck with it.

The body of the car is a plastic food container with one front and two side window that are crossed with copper wire. Inside of the container is a metal base plate that the rat stands on while driving. As you can see from the image above, the chassis has two horizontal panels.

Supplies

We started with an Elegoo Smart Car (V3) kit. You could use other kits or buy the parts separately.

Materials

Arduino Uno or other Arduino compatible board*

L298N Motor Driver Board*

Resistor in 1 M Ohm range (x3)

Small universal solder breadboard (a solderless breadboard is also useful for prototyping)

Chassis plates (x2)*

Motors (x4)*

Wheels (x4)*

Mounting plates for the motors (x4)*

3.7V lithium ion batteries (x2)*

Battery holder with switch*

Spacers (x6)*

Hex screws(M3 size)* ~1.25 in. long (x8); ~0.5 in. long (x22)

Hex nuts (x8)*

Electrical tape

Velcro tape

Plastic canister with removable lid

Insulated wire

Dupont connectors x8

Bare copper wire

Thin metal plate for floor board (we used aluminum because we had some lying around)

*indicates something we pulled from the Elegoo Inc. Smart Robot Car Kit.

Tools

Dremel or saw - for cutting plastic

Drill and Utility knife - for poking small holes in plastic

File - for smoothing out sharp edges

Pliers or small wrench
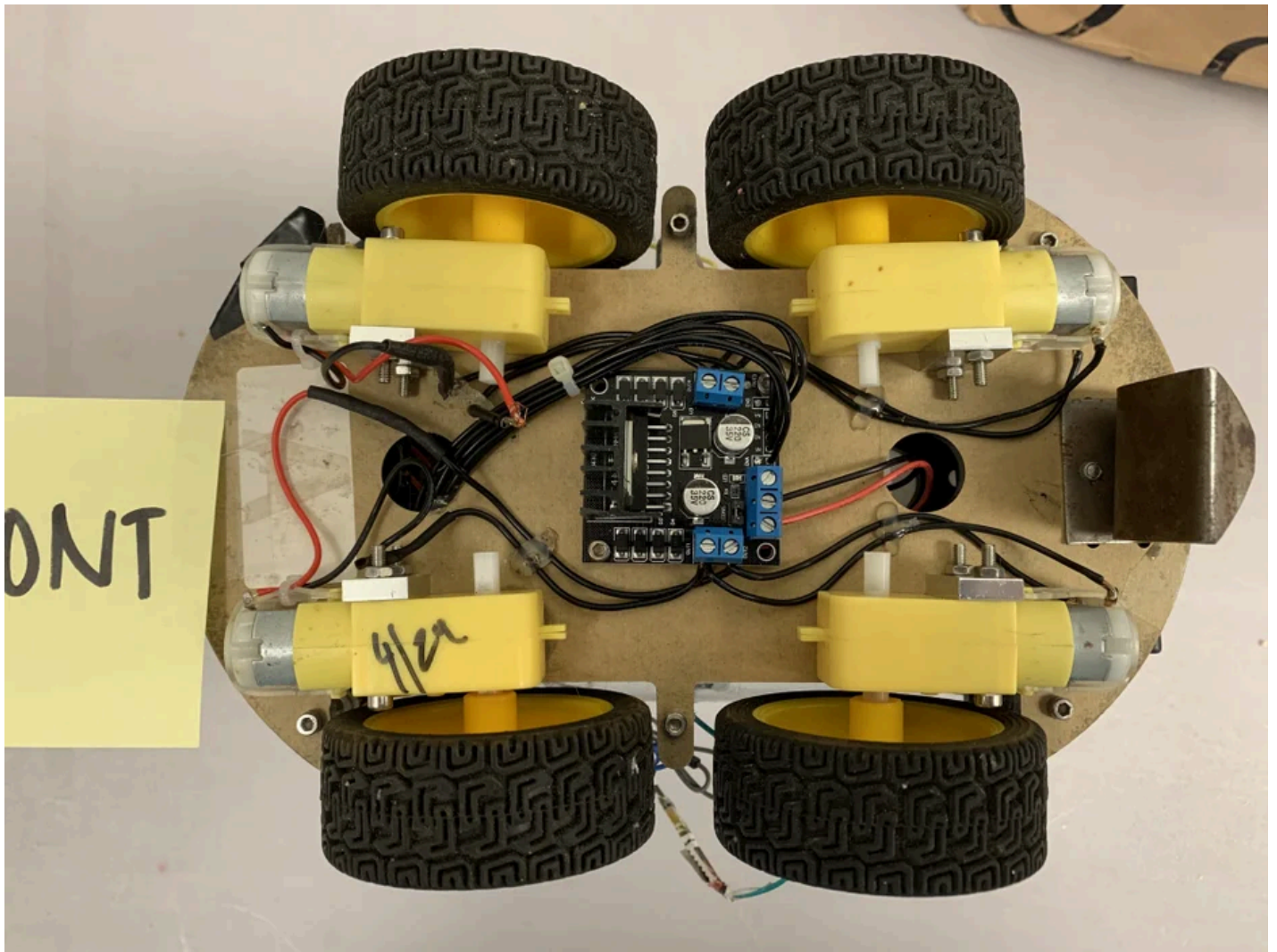
Screwdrivers

Wire stripper/cutter

Multimeter (for troubleshooting)

Soldering kit (with soldering iron, solder, and solder wick or bulb)

Software

Arduino IDE

Step 1: Assemble the Underside of the Car's Lower Panel.

Following instructions from the Elegoo kit, mount motors, wheels and motorboard on the underside of the car. (Note: If later you find that one of your motors is spinning the wrong way, just switch the wires of that motor.)

Handy tips - Use a glue gun to attach any hanging wires to the bottom of the car. The motors won't last forever. Replacements can be found by searching amazon.com for 3-6V gear motor. The bolts holding the motors on will loosen with use. Tighten these up periodically. A spot of glue might help.

Step 2: Assemble the Topside of the Car's Lower Panel.
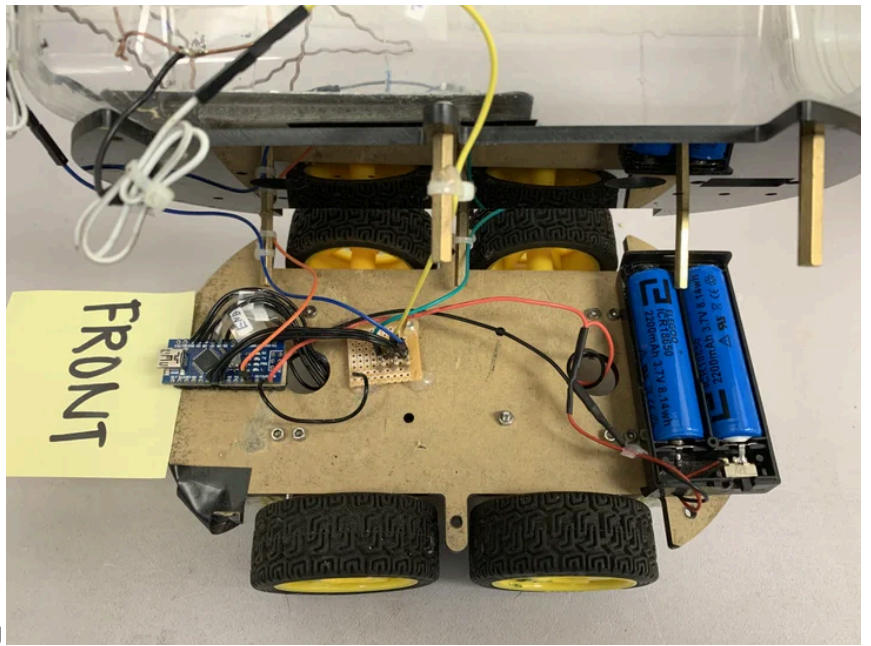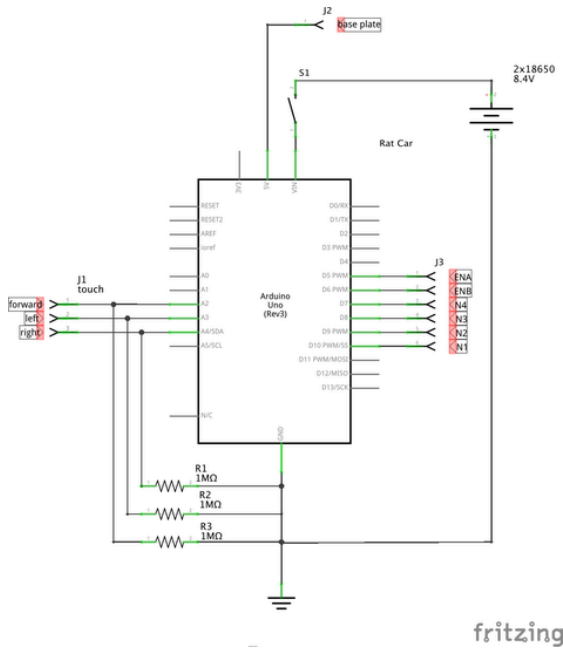
On the top side of the lower panel, at the back of the car, mount the battery holder. We used velcro tape and left a couple of inches slack in the wires so that we could easily pull out the battery and change the batteries.

We recommend prototyping first with a solderless breadboard. Once you have the car working, you'll want to use a solder breadboard for stability. Solderless connections and glued or taped connections are fine for getting started, but they will come apart with use.

Following the circuit diagram above:

1. Connect the Arduino to the motorboard.

2. Connect the batteries to both the Arduino and to the motorboard

3. Connect the Arduino to the motorboard

4. Mount three 10MOhm resistors to the breadboard

5. Connect the Arduino to the breadboard

6. Connect three insulated wires (each a different color) to the breadboard. (Later these will be attached to the car body.)

7. Connect an insulated wire from the Arduino's 5V pin (Later this will be attached to the metal base plate in the car)

Step 3: Install the Code

With the battery holder's power switch turned to "off," upload the code below to your arduino. Note that the code is commented to show you how you can adjust the speed and other parameters. Check the speed on your own car and make sure it is suitable for your rat. For safety and comfort, you don't want it to be too fast, but if it's set too slow and your rat is very heavy, the car may struggle to move. After you've installed the code, unplug the usb cable from the car.

```
// Rat Car
// originally by: LE Crawford, University of Richmond
// modified by: Thad Martin

volatile int state = LOW;
int EN_LEFT=6; //enable pin, ENB
int EN_RIGHT=5;  //ENA

int IN1=10;  //Direction pins left
int IN2=9;
int IN3=8;  //Direction pins right
int IN4=7;

int ABS_LEFT=150;  //Use an even number.
int ABS_RIGHT=150;  //Use an even number.
int CURRENT_LEFT=0;
int CURRENT_RIGHT=0;
int TARGET_LEFT = 0;
int TARGET_RIGHT = 0;
int TURN_OFFSET = 50;  // Adjust how sharp it turns.
                       // 50 makes for just a slight turn.  Much more, and the differences between left
                       // and right motors makes one turn direction sharper than the other.

int ACCELERATION = 10;  // 1 feels laggy, 50 feels lurchy.
bool LEFT_FORWARD = true;
bool RIGHT_FORWARD = true;

const byte forwardInputPin = A2;      // the forward touch sensor
const byte leftInputPin = A3;      // the left touch sensor
const byte rightInputPin = A4;      // the right touch sensor

int forwardInputVal = 0;
int leftInputVal = 0;
int rightInputVal = 0;

int threshold = 500;  // minimum ADC reading for touch sensor.

int moveState =
int oldMoveState = 0;

void setup()
{
  Serial.begin(9600);  // Serial monitor
  pinMode(IN1,OUTPUT);
  pinMode(IN2,OUTPUT);
  pinMode(IN3,OUTPUT);
  pinMode(IN4,OUTPUT);
  pinMode(EN_LEFT,OUTPUT);
  pinMode(EN_RIGHT,OUTPUT);
  pinMode(forwardInputPin, INPUT);
  pinMode(leftInputPin, INPUT);
  pinMode(rightInputPin, INPUT);
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);            // left wheels go forward
  digitalWrite(IN3,HIGH);
  digitalWrite(IN4,LOW);            // right wheels go forward
}

void loop()
{
  moveState = checkInputs();
  if (oldMoveState != moveState)
    setTarget(moveState);

  move();

  oldMoveState = moveState;
}

int checkInputs()
{
  // 0:stop, 1: forward, 2: turn left, 3: rotate left, 4: turn right, 5: rotate right.
  leftInputVal = analogRead(leftInputPin);
  delay(10);
  leftInputVal = analogRead(leftInputPin);

  forwardInputVal = analogRead(forwardInputPin);
  delay(10);
  forwardInputVal = analogRead(forwardInputPin);

  rightInputVal = analogRead(rightInputPin);
  delay(10);
  rightInputVal = analogRead(rightInputPin);

//  Serial.print("forward: ");
//  Serial.println(forwardInputVal);
//
//  Serial.print("left: ");
//  Serial.println(leftInputVal);
//
//  Serial.print("right: ");
//  Serial.println(rightInputVal);
//  delay(500);

  if(leftInputVal > threshold && forwardInputVal > threshold){
    moveState = 2;
  } else if (rightInputVal > threshold && forwardInputVal > threshold){
    moveState = 4;
  } else if (leftInputVal > threshold){
    moveState = 3;
  } else if (rightInputVal > threshold){
    moveState = 5;
  } else if (forwardInputVal > threshold){
    moveState = 1;
  } else {
    moveState = 0;  // Touching none is stop.
  }
  //moveState = 4;
  return (moveState);
}

void setTarget(int) // Set target speed for the wheels.
{
  // 0:stop, 1: forward, 2: turn left, 3: rotate left, 4: turn right, 5: rotate right.
```

```
  switch(moveState){
    case 0:
      TARGET_LEFT = 0;
      TARGET_RIGHT = 0;
      break;

    case 1:
      //Serial.println("forward");
      TARGET_LEFT = ABS_LEFT;
      TARGET_RIGHT = ABS_RIGHT;
      break;
```
case 2:
```
      TARGET_LEFT = ABS_LEFT – TURN_OFFSET;
      TARGET_RIGHT = ABS_RIGHT;
      break;
```
case 3:
```
      TARGET_LEFT = ABS_LEFT * –1;
      TARGET_RIGHT = ABS_RIGHT;
      break;
```
case 4:
```
      TARGET_LEFT = ABS_LEFT;
      TARGET_RIGHT = ABS_RIGHT – TURN_OFFSET;
      break;
```
case 5:
```
      TARGET_LEFT = ABS_LEFT;
      TARGET_RIGHT = ABS_RIGHT * –1;
      break;
  }
}
```
void move()
```
{
  if(CURRENT_RIGHT < TARGET_RIGHT){
    CURRENT_RIGHT +=ACCELERATION;
  }
  if(CURRENT_RIGHT > TARGET_RIGHT){
    CURRENT_RIGHT –=ACCELERATION;
  }
```
if(CURRENT_LEFT < TARGET_LEFT){
```
    CURRENT_LEFT +=ACCELERATION;
  }
  if(CURRENT_LEFT > TARGET_LEFT){
    CURRENT_LEFT –=ACCELERATION;
  }
```
// If necessary, switch the left wheels direction.
```
  if(CURRENT_LEFT < 0 && LEFT_FORWARD){
      digitalWrite(IN1,LOW);
      digitalWrite(IN2,HIGH);         // left wheels go backward
      LEFT_FORWARD = false;
  }
  if(CURRENT_LEFT > 0 && !LEFT_FORWARD){
      digitalWrite(IN1,HIGH);
      digitalWrite(IN2,LOW);          // left wheels go forward
      LEFT_FORWARD = true;
  }
```
// If necessary, switch the right wheels direction.
```
  if(CURRENT_RIGHT < 0 && RIGHT_FORWARD){
      digitalWrite(IN3,LOW);
      digitalWrite(IN4,HIGH);          // right wheels go backward
      RIGHT_FORWARD = false;
  }
  if(CURRENT_RIGHT > 0 && !RIGHT_FORWARD){
      digitalWrite(IN3,HIGH);
      digitalWrite(IN4,LOW);          // right wheels go forward
      RIGHT_FORWARD = true;
  }
```
analogWrite(EN_RIGHT, abs(CURRENT_RIGHT));
```
  analogWrite(EN_LEFT, abs(CURRENT_LEFT));

  //delay(2);  // enough delay from sensors.
  // Serial.print("left");
  // Serial.println(CURRENT_LEFT);
  // Serial.print("right");
  // Serial.println(CURRENT_RIGHT);
}
```

Or upload from here: https://github.com/LElizabethCrawford/RatCarCode/b...

Step 4: Test It Out

Charge up your batteries, turn on the car and test it out. You should now be able to drive the car by touching the exposed end of the 5V wire to each of the colored wires coming off of the circuit board. You should also be able to close the circuits by holding the 5V wire in one hand and touching each of the other wires with your other hand to control the car. (If your skin is dry or calloused, you might not be conductive enough. Try moisturizing. Or use your palms rather than fingertips.)

When you do this, a small current flows from the Arduino, through your body and back to the arduino. You should not be able to feel it. Make note of which wires control forward, left, and right. Label these wires. For now, tuck the left and right wires out of the way and focus on the forward wire.

A note on the code: Depending on the hardware you use, this line may need to be adjusted

```
  int threshold = 500;  // minimum ADC reading for touch sensor.
```

If the car moves without touching the wires, decrease the threshold. If you touch the wires and it does not move, increase it. The range is 0 to 1024, perhaps increment it by 100 to find a suitable value.

Or uncomment the lines:

```
// Serial.print("forward: ");
// Serial.println(forwardInputVal);
```

To see what the values are when you are touching and not touching the wires.

Step 5: Install the Top Panel of the Chassis.

Using the spacers and hex screws, install the top panel of the chassis.

Step 6: Create the Driver's Compartment.

1. Take the plastic food container and cut out a front window. (At first we used a dremel, which melted the plastic. You might try something prettier.) The exact size and placement of the windows will depend on your animal and the specific canister you chose. The idea is to allow your rat to see and smell out the front of the car while maintaining the structural integrity of the canister so that you can install side windows later. For our canister, the front window is about 4" x 3.5".

Handy tip. You may want to start only with the front window and get your rat comfortable driving forward. You can always add the side windows and steering options later.

2. File down any sharp edges on the plastic.

3. Decide on the placement of your bars. We recommend having several horizontal bars at different heights so that your rat can find the most comfortable height from a crouching or standing posture. (Also, plan for rat pups to grow -- they may use a low bar at first and later use a higher one.) We made the bars about 1" apart.

Handy tips: Horizontal bars are easier to grasp than vertical ones. Also, you want wire that is easy enough to bend, but not so fine that it is uncomfortable for the rat to grasp. Finally, for safety, you want your rat to enter and exit through the back of the container and not crawl between the bars. If it crawls between the bars, the car might start driving forward while the rat is halfway out of the car.

4. Thread the wire through. It is convenient to make several bars out of one continuous piece of wire. Make sure you leave no sharp points that could hurt you or your rat.

5. Connect an insulated wire to your copper bars that will eventually connect to the forward controlling wire that comes off of the breadboard.

Handy tip: Make this wire the same color as the forward controlling wire so it will be easy to match them up later. (We wish we had done this.)

6, Create a metal floor plate that is big enough to fill up most of the floor of the container, but small enough to fit through the opening. We used aluminum. Solder a wire to the underside of the floor plate.

Handy tip: Make this wire the same color as the one coming off of the Arduino's 5V so it will be easy to match them up later.

7. Drill a hole in the side of the container near the floor through which you can thread the wire from the floor plate. *Handy tips: Don't make this hole on the floor of your canister -- if your rat pees, you'll want that to be contained. Also, eventually you'll put a connecter on the end of that wire. Cleaning will be easier if the hole is big enough for the connector to pass through.*

8. Use velcro to attach the metal plate to the bottom of the container. You want the floor plate to be stable, but also easily removable for cleaning.

Step 7: Assemble and Test

1. Using velcro tape, attach the container to the top of the top panel.

2. Twist together the end of the wire coming off of the base plate to the wire coming off of the Arduino's 5V.

3. Twist together the end of the wire coming off of the front window to the forward going wire coming off of the bread board.

4. Turn on the car (make sure batteries are charged).

5. Gently touch your palm on the base plate and touch the front bars with your fingers. The car should go forward when you make contact with bars and stop when you release contact.

6. Once that's working, replace the twists you did with connectors. We used dupont connectors, but a sturdier connector might work better.

Step 8: Be Safe Out There

Here is some general safety advice.

1. The rat's paws, mouth, and nose are extremely sensitive. Check very carefully for any sharp or pointy pieces that could cause injury. File these and cover them up with tape or plastic.

2. Once your rat is driving, make sure it can't get hurt by driving into things. Arrange the container and wires so that there is a little space between the rat and the frontmost part of the car. That way, if it hits a wall, the front of the car will make contact but the rat will not. We recommend setting up a driving arena with walls that give.

3. Practice good hygiene. After training sessions, clean any car parts that the rat touches.

4. Always supervise your rat very, very closely. It might start chewing on the wires and it could hurt itself. It might do something else you did not expect. Watch closely for any sign of stress or discomfort.